

Deteksi Kendaraan Bergerak Secara *Real Time*

Riza Ansori¹, Ir. Dadet Pramadihanto, M.Eng, PhD², Nana Ramadijanti, S.Kom, M.Kom²
Mahasiswa¹, Dosen²

Politeknik Elektronika Negeri Surabaya
Institut Teknologi Sepuluh Nopember
Kampus PENS-ITS Keputih Sukolilo Surabaya 60111
Telp (+62)31-5947280, 5946114, Fax. (+62)31-5946114
Email : ansori.eepis@gmail.com

Abstrak

Kendaraan bergerak merupakan suatu objek menarik untuk dilakukan suatu penelitian. Hal-hal yang menarik yang dapat dijadikan objek penelitian adalah kecepatan, jenis dan warna dari kendarannya. Kendaraan yang bergerak dengan cepat dapat menimbulkan blur yang dinamakan *Motion Blur Image*. *Motion Blur Image* adalah efek yang disebabkan oleh gerakan relatif antara kamera dan objek yang bergerak. Umumnya kita akan menghindari efek ini agar foto yang kita harapkan mendapatkan hasil yang bagus. Dalam proyek ini motion blur image kita gunakan untuk mendeteksi kecepatan dari suatu kendaraan sekaligus mengetahui jenis dan warna kendaraan tersebut. Untuk memperkirakan kecepatan kendaraan kita analisa hasil dari motion blur dan kita cocokkan dengan menggunakan gabor filter dan *template matching* dari sample kendaraan yang telah kita tentukan

Kata Kunci : Motion Blur Image, filter gabor, template matching

1. Pendahuluan

1.1 Latar Belakang

Motion Blur Image adalah efek yang disebabkan oleh gerakan relatif antara kamera dan objek yang bergerak. Umumnya kita akan menghindari efek ini agar foto yang kita harapkan mendapatkan hasil yang bagus. Dalam proyek ini motion blur image kita gunakan untuk mendeteksi kecepatan dari suatu kendaraan sekaligus mengetahui jenis dan warna kendaraan tersebut.

Kebanyakan metode yang digunakan untuk mengetahui kecepatan kendaraan menggunakan LIDAR (Laser Infrared Detection And Ranging) atau RIDAR (Radio Detection And Ranging) yang digunakan untuk mengurangi pelanggaran lalu lintas batas kecepatan kendaraan. Tetapi metode ini membutuhkan biaya yang besar untuk membeli alatnya. Sebagai alternatif, kita menggunakan motion blur image untuk memperkirakan kecepatan suatu kendaraan yang sedang melintas di jalan raya.

Untuk memperkirakan kecepatan kendaraan kita analisa hasil dari motion blur dan kita cocokkan dengan menggunakan line matching dari sample kendaraan yang telah kita tentukan

1.2 Rumusan Permasalahan

Adapun permasalahan yang ada pada sistem teknologi ini yaitu sebagai berikut :

1. Mendapatkan gambar dari kendaraan di jalan raya.
2. Menghitung panjang dari blur sebagai parameter menghitung kecepatan.
3. Mengetahui kecepatan, jenis dan warna dari gambar yang didapatkan.

1.3 Tujuan Proyek

Tujuan proyek akhir ini adalah tujuan proyek akhir ini adalah mendeteksi kecepatan kendaraan sekaligus mengetahui warna dan jenis dari kendaraan yang diambil menggunakan sebuah kamera yang menghasilkan gambar blur.

1.4 Kontribusi Proyek

Proyek Akhir ini nantinya diharapkan dapat dikembangkan untuk memberikan analisa yang pada akhirnya memberikan informasi yang berguna tentang kendaraan yang melaju di jalan.

2. Teori Penunjang

Motion blur merupakan akibat dari gerakan relative antara kamera dan objek yang dicapture . biasanya digunakan untuk

- Restorasi gambar
- Analisa motion
- Meningkatkan resolusi dari video
- Efek special
- Memperkirakan kecepatan

2.1. Degradasi Image

Degradasi image adalah space invariant linear system yang dicirikan oleh point spread function (PSF) $h(x,y)$ dalam kasus uniform linear motion

$$g(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x-\alpha, y-\beta) f(\alpha, \beta) d\alpha d\beta \quad (2.1)$$

Degradasi dalam uniform linear motion:

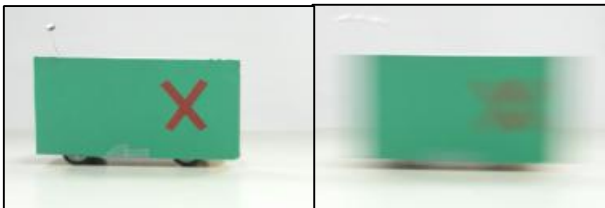
$$h(x,y) = \begin{cases} \frac{1}{R}, & |x| \leq \frac{R}{2} \cos \theta, y = x \tan \theta \\ 0, & \text{otherwise} \end{cases} \quad (2.2)$$

Di mana θ dan R adalah arah motion dan panjang dari motion blur. Untuk merestorasi gambar asli dengan memperkirakan $h(x,y)$ atau equivalent θ dan R dari blur image $g(x,y)$. Ada dua varian yaitu :

- Partial blur adalah blur yang disebabkan oleh campuran dari objek yang bergerak dan background masih berada dalam sekitar batas objek
- Total blur adalah blur yang dipaksa oleh gerakan objek di dalam wilayah

2.2. Blur Parameter Estimation

Untuk motion blur image dengan objek yang bergerak dalam scene yang statis , blur parameter diperkirakan dari posisi awal dan panjang dari partial blur sepanjang arah gerakan . Seperti yang ditunjukkan oleh gambar di bawah ini :

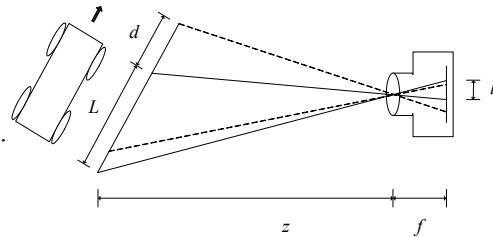


Gambar 1. Gambar objek statis dan bergerak

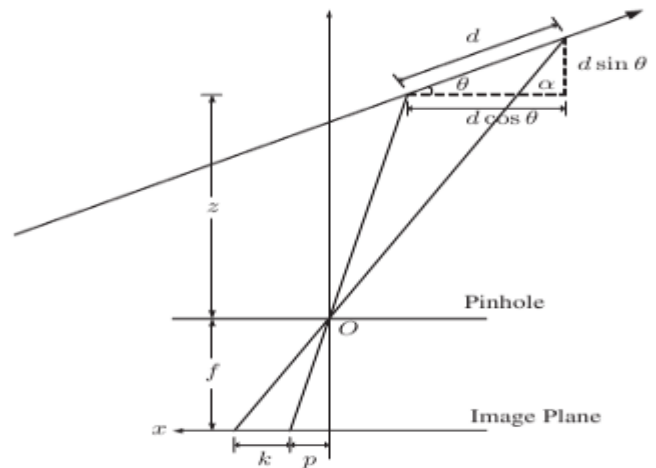
Untuk objek static dengan sharp edges dengan intensitas dalam scanline blur dari daerah partial blur dalam motion blur image yang tersebar di sejumlah pixel dan dapat menjadi model a ram edge di intensitas profil dari gambar scanline. Jadi ada 2 edge detection yang digunakan yaitu :

- 1) Sharp edge \rightarrow step response
 - 2) Blur edge \rightarrow ramp response
- Parameter intrinsic:
 - a. Panjang focal (Focal Length) dari kamera
 - b. Ukuran size CCD pixel kamera
 - c. Exposure Time
 - Parameter Ekstrinsik
 - a. Jarak dengan objek
 - b. Orientasi kamera saat pengambilan gambar

2.3. Formula Untuk Menghitung Kecepatan



Gambar 4. Pinhole model untuk general case



Gambar 5. Model kamera untuk estimasi kecepatan kendaraan

Metode untuk mendeteksi kecepatan kendaraan berdasarkan prinsip model pinhole kamera. Seperti yang ditunjukkan pada gambar 5 kita andaikan sudut antara arah gerak objek dan kamera adalah θ dan perpindahan objek adalah d dalam interval waktu tertentu, kemudian kita dapatkan

$$\frac{a}{p+k} = \frac{d \sin \theta}{f} \dots\dots\dots(2.3)$$

dan

$$\frac{d \cos \theta - \alpha}{k} = \frac{z}{f} \dots\dots\dots(2.4)$$

dimana p dan k adalah posisi awal dan perpindahan objek dalam kamera, kemudian kita dapatkan

$$d = \frac{zk}{f \cos \theta - (p+k) \sin \theta} \dots\dots\dots(2.5)$$

di mana z adalah jarak antara objek dan kamera dalam arah sejajar dengan sumbu X dan f adalah focal length dari kamera, Apabila exposure time dari kamera adalah T dan ukuran pixel CCD kamera adalah s_x . Kemudian kecepatan v didapatkan

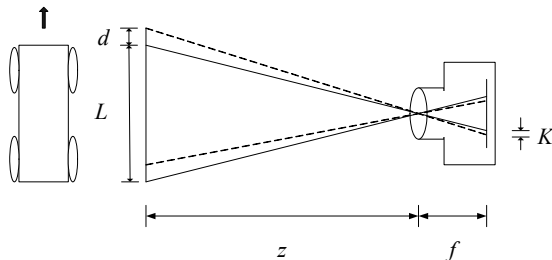
$$v = \frac{d}{T} = \frac{zKs_x}{T[f \cos \theta - s_x(P+K) \sin \theta]} \dots\dots\dots(2.6)$$

kemudian kita sederhanakan, sehingga menjadi

$$v = \frac{zKs_x}{Tf \cos \theta} \dots\dots\dots(2.7)$$

dalam formula di atas, z dan s_x merupakan parameter internal dari kamera. T merupakan exposure time(shutter speed) yang didapatkan saat pengambilan gambar. Kemudian untuk memperkirakan kecepatan objek menggunakan motion blur image, kita sertakan parameter yang telah diidentifikasi dari gambar (panjang blur K dan posisi objek P) dan posisi relatif dan orientasi antara objek dan kamera (jarak z dan sudut arah gerak objek θ).

Untuk kasus seperti gambar di bawah ini, dimana objek bergerak secara paralel terhadap image plane dari kamera



Gambar 6. Pinhole model untuk special case

dan untuk mendapatkan kecepatan digunakan rumus dapat kita sederhanakan menjadi

$$v = \frac{zKs_x}{Tf} \dots\dots\dots(2.8)$$

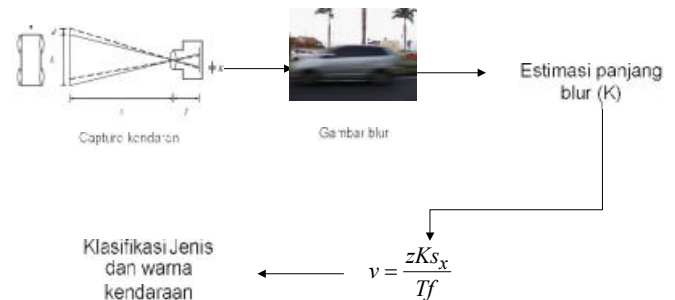
$$z = \frac{fL}{s_x l} \dots\dots\dots(2.9)$$

Rumus di atas digunakan jika mengetahui jarak dari objek ke kamera, jika tidak diketahui secara pasti kita dapat menghitung jaraknya dengan

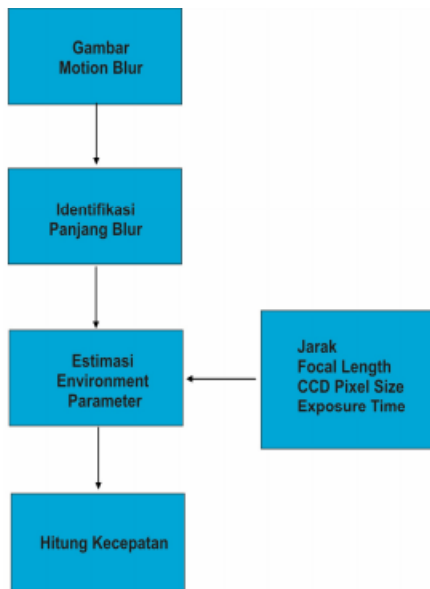
Di mana L adalah panjang dari kendaraan yang telah diketahui (biasanya dari data manufaktur) dan l adalah panjang kendaraan dalam gambar. f focal length kamera dan s_x pixel size CCD kamera. Berdasarkan formula ini kecepatan kendaraan dapat dihitung dengan :

$$v = \frac{KL}{Tl} \dots\dots\dots(2.10)$$

3. Rancangan Sistem



Gambar 7. Blok diagram sistem



Gambar 8. Blok diagram penghitungan kecepatan kendaraan

3) Capture target

Proses capture adalah proses pengambilan kendaraan melalui melalui kamera. Pengambilan gambar ini dilakukan di jalan . Agar gambar stabil maka digunakan tripod untuk mencegah pergeseran posisi pengambilan gambar. Sebelum dilakukan pengcaptur-an, perlu dilakukan persiapan terlebih dahulu seperti estimasi jarak pengambilan objek dari kamera dan dilakukan *setting* terhadap kamera, yaitu *mode* pengambilan gambar diset ke *shutter priority* agar kita dapat dengan mudah menetapkan *exposure time/shutter speed*. Settingan ini penting sekali, kaena diperlukan dalam penghitungan kecepatan nantinya. Untuk *shutter speed* kita set 1/50 sec

4) Estimasi panjang blur

Untuk mengestimasi panjang blur , langkah-langkahnya sebagai berikut :

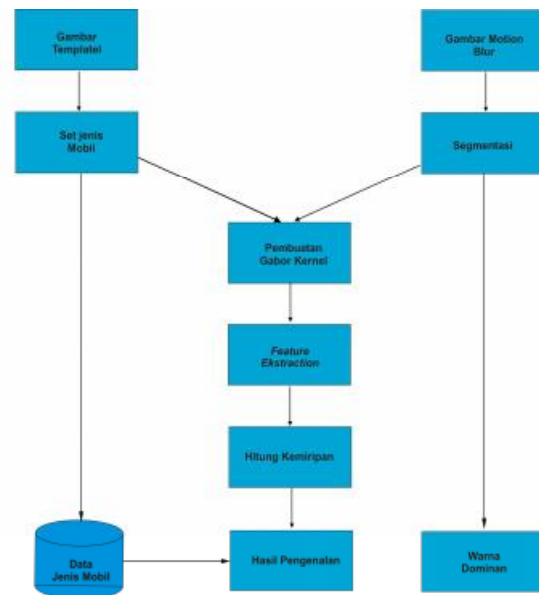
- ❖ Hitung Fourier Transfor
- ❖ Hitung Log Spectrum
- ❖ Lakukan Inverse Fourier
- ❖ Hitung rata-rata piksel berdasarkan kolom
- ❖ Mencari nilai negatif pertama di bagian *real part*, jika tidak ada maka diambil yang terkecil nilainya. Nilai yang didapatkan ini dicari pada kolom berapa, yang merupakan korespondens panjang blur

5) Estimasi environment parameter

Selain parameter dari blur image, untuk menghitung kecepatan kendaraan digunakan parameter lainnya seperti jarak objek dengan kamera, orientasi pengambilan gambar, focal length, exposure time, dan pixel size CCD kamera.

6) Menghitung kecepatan

Setelah mendapatkan parameter-parameter yang diperlukan seperti panjang blur, jarak objek, orientasi/sudut pengambilan gambar, focal length dan, exposure time/shutter speed dan pixel size CCD kamera, kemudian dilakukan penghitungan kecepatan, sehingga didapatkan kecepatan kendaraan yang terdeteksi



Gambar 9. Blok diagram penentuan jenis kendaraan

1) Capture target

Proses capture adalah proses pengambilan kendaraan melalui melalui kamera. Pengambilan gambar ini dilakukan di jalan . setelah itu didapatkan gambar blur. Blur ini didapatkan karena pergerakan target yang sangat cepat.

2) Segmentasi

Pada tahap ini dilakukan beberapa tahap untuk memisahkan objek dari backgroundnya, adapun tahap-tahap yang dilakukan sebagai berikut :

- Load gambar foreground dan background lalu ubah menjadi grayscale. Dalam melakukan konversi dari gambar berwarna ke gambar grayscale, maka

sistem 5870 dan channel B dikalikan dengan 0.1140.

-Lakukan *background subtraction*. Pada tahap ini image yang dianggap sebagai background adalah image yang pertama kali ditangkap. Tujuan dari *background subtraction* adalah untuk melakukan pendeteksian apakah ada objek atau tidak. Filter Blur

- Setelah dilakukan *background subtraction*, maka gambar hasil perlu dilakukan *smoothing* untuk memperhalus hasil dan mengurangi noise yang ditimbulkan

-Lalu kita lakukan *cropping* terhadap gambar hasil *subtraction* tadi, proses ini. Sebelum dilakukan *cropping*, dilakukan pendeteksi tepi untuk menemukan tepi-tepi dari objek tersebut. Untuk deteksi tepi digunakan *canny edge detection*. Setelah itu kita cari batas atas, kanan, bawah dan kiri dari objek tadi.. setelah ditemukan batas-batas tersebut, kita lakukan *cropping* pada batas tadi.

3) Pembuatan Gabor Kernel

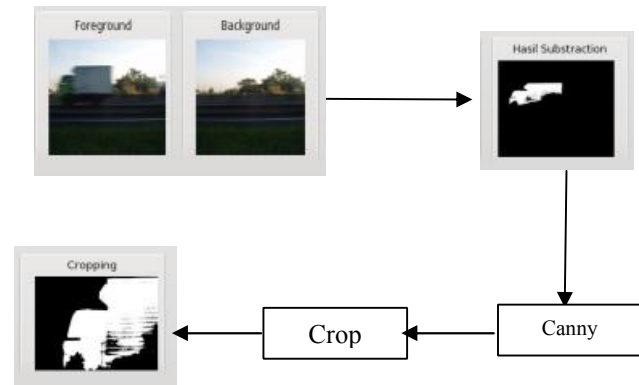
Pada tahap ini, kita membuat gabor kernel terlebih dahulu, dalam pembuatan kernel ini dengan parameter scale sebanyak 4 dan orientasi sebanyak 5 sehingga total kernel yang dibuat sebanyak 20 kernel

4) Feature Ekstraksi




Pada tahap ini kita lakukan ekstraksi fitur template dari masing –masing jenis mobil yang telah kita set dan menkonvolusikan dengan kernel yang telah kita buat. Begitu juga untuk fitur dari gambar uji

5) Deteksi Warna

Kemudian dilakukan ekstrak warna dari gambar objek, sehingga kita dapat kan RGB tiap pixel. dan dapat mengetahui warna dari kendaraan tersebut



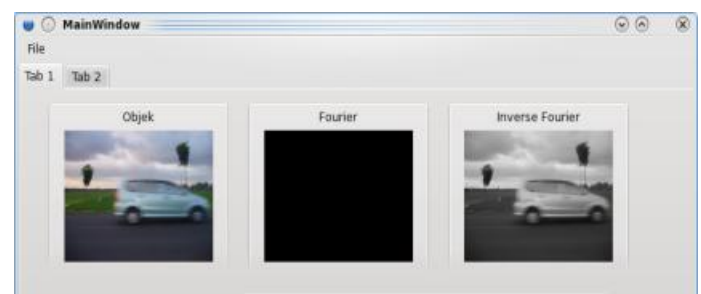
Gambar 10.. Proses Segmentasi

Gambar Asli	Gabor Kernel	Hasil Konvolusi
		

Gambar 11.. Proses Feature Ekstraksi

4. Hasil dan Pembahasan

4.1 Hitung Kecepatan



Gambar 12. Proses menghitung panjang blur



Gambar 13. Hasil Estimasi Kecepatan

Daftar Pustaka

[1] Huei-Yun Lin. 2005. Vehicle Speed Detection and Identification from a Single Motion Blurred Image. Proceedings of the Seventh IEEE Workshop on Applications of Computer Vision (WACV/MOTION'05)

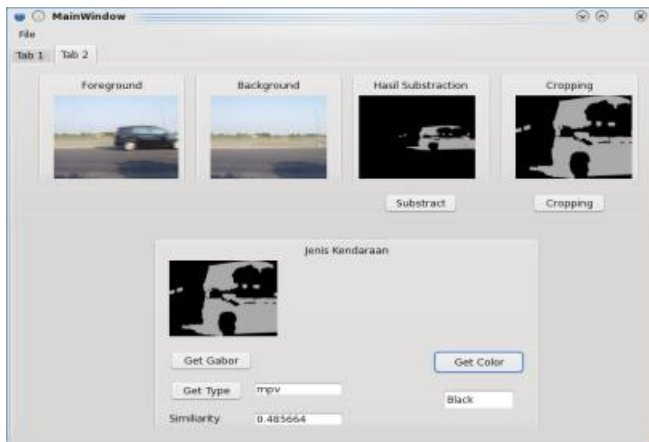
[2] R.Lokhande,K.V.Arya.2006.Identification of Parameters and Restoration of Motion Blurred Images.

[3] Thiang,Teguh Hantono Andre .Type of Vehicle Recognition Using Gabor Filter Representation and Template Matching Method.

[4] Hartono,Fransisco ,Jimmy.2008.Perancangan dan Pembuatan Aplikasi Pengenalan Jenis Mobil dengan Menggunakan Gabor Filter dan Jaringan Syaraf Tiruan.

[CV Penulis]

Riza Ansori menjalankan studi D4 bidang Teknik Informatika pada Politeknik Elektronika Negeri Surabaya – Institut Teknologi Sepuluh Nopember(PENS-ITS) semester 8.



Gambar13. Hasil Deteksi Tipe dan Warna

5. Hasil Dan Kesimpulan

Dari hasil uji coba perangkat lunak ini dapat ditarik beberapa kesimpulan :

- ❖ Hasil pendeteksian dalam system ini sangat dipengaruhi oleh pencahayaan
- ❖ Proses segmentasi dalam system ini sangat tergantung .dengan background yang diambil , semakin mirip dengan dengan foreground semakin baik.Sehingga pendeteksian tipe menjadi semakin tepat.